基于申威众核处理器的 NSGA-II 并行和优化方法 *

刘 垚[1,2]，郑 琳[1,2]，郑 凯[1,2]，王 肃[1,2†]，廖启丹[1,2]

(1. 华东师范大学 计算机科学与软件工程学院，上海 200062; 2. 数学工程与先进计算国家重点实验室，江苏 无锡 214215)

摘 要：由申威众核处理器组成的“神威·太湖之光”是当前我国性能最高的超级计算机，可为大规模 NSGA-II 求解提供硬件平台。基于硬件架构特点，设计了“分岛—主从增强”混合并行 NSGA-II。在主从模式基础上，利用从核间寄存器通信，实现核组内从核局部数据存储的共享。优化流程，实现更多算法模块在从核上的并行。运用 DMA 传输、向量化、双缓冲、存储优化等方法显著提高加速比。实验表明，优化的并行 NSGA-II 在申威众核处理器上具有良好的加速比和扩展性。

关键词：申威众核处理器；NSGA-II；并行遗传算法；多目标；并行优化

中图分类号：TP338.6 doi: 10.19734/j.issn.1001-3695.2018.06.0511

Parallelization and optimization of NSGA-II based on sunway many-core processor

Liu Yao[1,2], Zheng Lin[1,2], Zheng Kai[1,2], Wang Su[1,2†], Liao Qidan[1,2]

(1. *School of Computer science & software Engineering*, *East China Normal University*, *Shanghai* 200062, *China*; 2. *State Key Laboratory of mathematical Engineering & Advanced Computing*, *Wuxi Jiangsu* 214215, *China*)

Abstract: The Sunway TaihuLight, which is composed of Sunway many-core processors, is currently the highest performance supercomputer in China. It can provide a hardware platform for NSGA-II to solve the large-scale problems. Considering the architecture of Sunway many-core processor, an “island combined with enhanced master-slave” hybrid parallel NSGA-II algorithm is designed. Based on the master-slave mode, register communication is used to realize the sharing of local data memory of CPEs in a core group. The algorithm process is optimized and more algorithm modules are parallelized on CPEs. By means of DMA transmission, vectorization, double buffering and storage optimization, the speedup is significantly increased. The experiments show that the optimized parallel NSGA-II has good speedup and scalability on the Sunway many-core processors.

Key words: sunway many-core processor; NSGA-II; parallel genetic algorithm; multi-objective; parallel optimization

0 引言

“神威·太湖之光”是世界上首台运算速度超过 10 亿亿次的超级计算机[1]，峰值性能达到 125.4 PFlops，由我国自主研发的申威众核处理器为核心构建而成。近年来，针对申威众核处理器的研究日益增多，基于其上的并行化实例也在日渐涌现[2~8]。洪文杰等人[3]对共性数学库 PETSc 在申威众核处理器上进行了并行，优化后最终的从核加速比达到 16.4 倍。孟德龙等人[4]将流体力学开源软件包 OpenFOAM 移植到申威众核处理器上，解决了兼容性问题和稀疏矩阵的计算开销较大的问题，单核组性能优化后加速比达到 8.03 倍。赵美婷等人[5]将广泛应用与行人检测的 HOG 特征提取算法进行加速，处理高分辨率多幅图像的从核加速比达到 83 倍。姚文军等人[6]将纳米级粒度分子动力学软件 NAMD 移植到神威太湖之光上，并行优化后性能提高了 20 倍。然而针对遗传算法在申威众核处理器上的并行化研究依然较少。

遗传算法是一种通过模拟自然进化过程搜索最优解的方法，根据目标函数的数量可分为单目标遗传算法[10~13]和多目标遗传算法[14~17]两种类型。针对单目标遗传算法，赵瑞祥等人[18]提出了基于申威众核处理器的单目标遗传算法的二级并行，按核组进行分岛，在从核上实现了适应度函数计算的并行，其从核加速比在重载的情况下最高达到 31.85 倍，然而没有解决在轻

收稿日期：2018-06-12；修回日期：2018-07-06 基金项目：数学工程与先进计算国家重点实验室开放基金资助项目（2016A05）

作者简介：刘垚（1981-），男，安徽灵璧人，高级工程师，硕士，主要研究方向为计算机网络、高性能计算；郑琳（1993-），女，浙江绍兴人，硕士研究生，主要研究方向为计算机网络、高性能计算；郑凯（1968-），男，浙江宁波人，副教授，博士，主要研究方向为计算机网络、云计算；王肃（1980-），女（通信作者），河南许昌人，讲师，博士，主要研究方向为智能优化、高性能计算（swang@cc.ecnu.edu.cn）；廖启丹（1991-），男，江苏盐城人，硕士研究生，主要研究方向为计算机网络、高性能计算.

载的情况下加速效果差的问题。多目标遗传算法方面，常见的多目标遗传算法有 VEGA[19]、MOGA[20]、NPGA-II[21]、NSGA-II[22,23]等。其中，NSGA-II 采用非支配排序的方法使算法具有逼近 Pareto 最优前沿的能力，采用拥挤度机制保证 Pareto 最优解能够良好的分布，但随着求解问题规模不断扩大和复杂程度不断增加，对于 NSGA-II 求解质量和运行速度都提出了更高的要求。沈焕学[24]将 NSGA-II 在申威众核处理器上实现了并行化，但在从核上只实现了目标函数计算部分的并行，导致轻载时从核加速比仅略高于 1。上述工作虽然在申威众核处理器上实现了基本的并行遗传算法，但是尚缺乏并行优化方法的研究。

基于申威众核处理器的体系架构特点，本文设计了一种"分岛—主从增强"混合式并行 NSGA-II 算法，利用 MPI 实现了核组间的分岛并行，利用 Athread 实现了核组内的主从并行。在主从模式的基础上，利用从核间寄存器通信，实现核组内从核局部数据存储的共享。通过对算法流程进行优化，实现了更多算法模块在从核上的并行，提高从核利用率。综合运用 DMA、向量化、双缓冲、存储优化等方法，显著提高了并行效率。利用分岛模型和迁移策略，将算法扩展到"神威·太湖之光"大规模集群上，证明了其良好的扩展性。最终，解决了轻载情况下加速效果较差的问题，保证了算法在多种负载情况下的较高加速比。

1 神威·太湖之光

"神威·太湖之光"计算机系统采用基于高密度弹性超节点和高流量复合网络架构和面向多目标优化的高效能体系结构，系统由 40 960 块"申威 26010"异构众核处理器（SW26010）组成，通过计算机插件板、计算超节点和计算机仓等模式进行系统扩展，构成高速计算系统[1]。

SW26010 采用片上计算阵列集群和分布式共享储存结合的异构众核体系结构，单处理器芯片集成 4 个运算核组共 260 个运算核心，每个核组包含 1 个运算控制核心（主核）和 1 个运算核心阵列，如图 1 所示。采用寄存器级数据通信、多模式异步数据流传输和运算阵列快速同步等技术提高运算核心协同执行效率。每个众核处理器配置 32 GB 内存，每核组本地内存为 8 GB；运算核心可直接离散访问主存，也可以通过 DMA 方式批量访问主存，运算核心阵列中同行与同列的核间可采用寄存器通信方式进行通信；每个运算核心的局部数据存储（LDM）大小为 64 KB，指令存储空间为 16 KB。

2 NSGA-II 的并行和优化

2.1 并行架构

NSGA-II 是基于非支配排序的多目标遗传算法，与传统单目标遗传算法的主要区别是在选择算子执行之前根据个体之间的支配关系进行了分层，并根据层次关系对每一个个体进行拥挤度计算。NSGA-II 的选择算子、交叉算子和变异算子与传统单目标遗传算法基本相同。

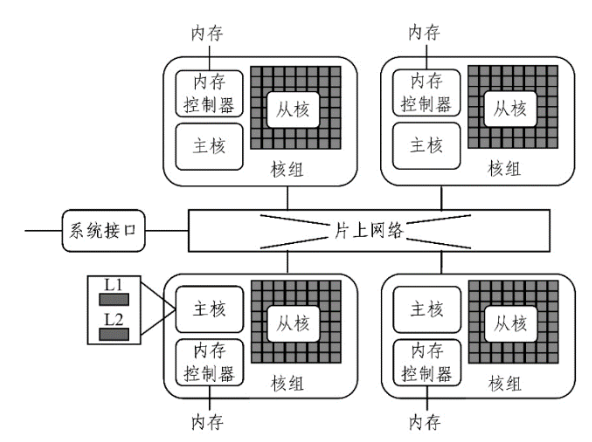


图 1 SW26010 异构众核处理器架构

Fig.1 Architecture of SW26010 heterogeneous many-core processor

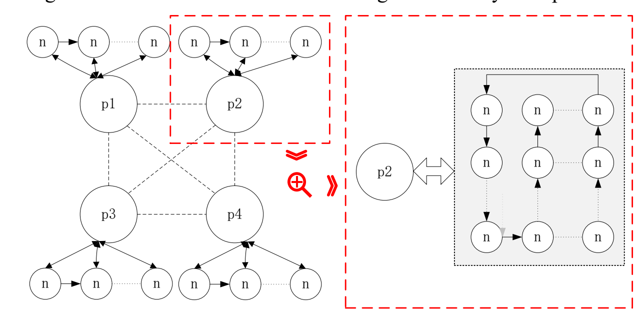


图 2 分岛—主从增强"混合式并行架构

Fig.2 Hybrid parallel architecture of island-enhanced master/slave

基于申威众核处理器的架构，本文设计了"分岛—主从增强"混合式并行 NSGA-II 算法，如图 2 所示。其中 p1、p2、p3、p4 分表代表四个子种群，每个子种群代表一个岛。各岛并行进化，并每隔一定代数进行迁移。每一子种群又按照分配的从核数量进行分组，组内的个体按主从式模型并行进化。同时，在从核计算时，各从核间按照一定规则进行数据通信来保证计算过程的完整性，因此是一种增强式的主从并行模式。各岛内，初始化、选择、交叉以及变异操作在主核中执行；从核除了负责目标函数计算，还负责非支配排序和拥挤度计算，这样可以充分利用从核的超高计算能力。由此实现了 NSGA-II 的二级混合式并行。

2.2 单核组并行

单个核组对应于一个岛，核组内的并行采用"主从增强"并行模式。因为目标函数计算的并行性能依赖于解决问题的计算量和数据量，在计算量和数据量较小的轻载情况下往往加速比较低，因此，本文对 NSGA-II 的更多操作模块实现并行。除了目标函数计算之外，非支配排序和拥挤度计算的过程占算法总体开销的比重较大，因此设计将这两部分实现并行加速。根据 SW26010 单核组（主核+从核阵列）的异构体系结构，将 NSGA-II 算法的主体计算代码进行单核组并行化，流程如图 3 所示。

加速线程库（Athread 库）[9]是针对主从加速编程模型所设计的程序加速库，其目的是使用户能更方便、快捷地对和核组内的线程进行控制和调度，从而更好地发挥核组内多计算核的计算性能。

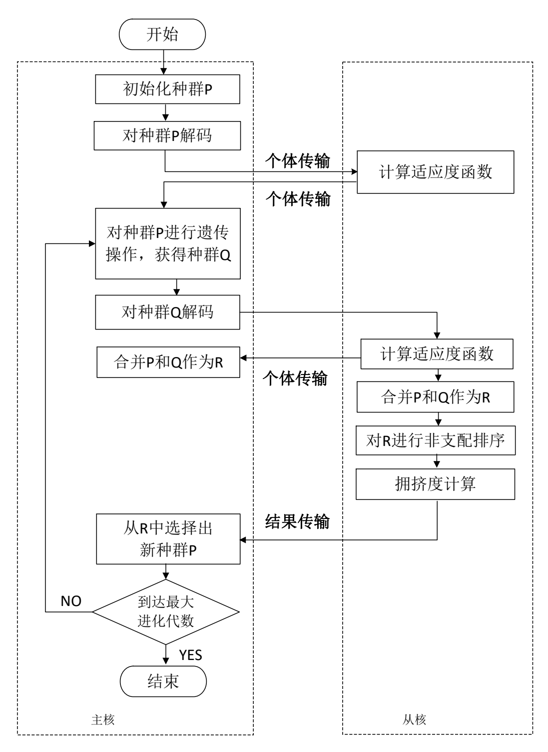


图 3　并行 NSGA-II 流程

Fig.3　Flow chart of parallel NSGA-II

在目标函数计算进行并行化时，由于每个个体的目标函数计算无相互关联，可将种群中的个体直接分成若干等份分别交给不同的从核进行计算。首先初始线程和创建线程组，启动核组中的 64 个从核资源，然后将种群内个体信息通过 DMA 方式加载到对应的从核上。从核对各个个体进行众核加速计算的同时主核处于等待的状态，计算完成后继续通过 DMA 方式将从核所得的目标函数值写回主核。

在非支配排序的过程中，每个个体必须与当前未被分层的所有个体进行比较，分层过程需要所有个体信息。由于从核只有 64 KB 局存，单个从核往往无法完整加载所有个体数据。如果从核同时进行不同个体的非支配排序，则从核需要以直接访问主存的形式读取种群中所有个体信息。这种方式在主从核间的通信开销很大，并行效果不佳，因此需要进一步的并行优化。

拥挤度计算以非支配排序的分层结果为基础。在计算前，首先在主核上对每一层的个体的目标函数值进行排序；然后将排序好的目标函数值交给对应从核，计算个体间目标函数的距离。计算完毕后，将个体所得各目标函数距离值传回主核作归一化和累加操作。这种方式的并行化程度很低，仍然有大部分运算在主核部分进行，也需要进一步的并行优化。

2.3　并行优化

在 NSGA-II 并行化过程中遇到以下挑战：a）目标函数并行时，在函数时间复杂度固定的情况下要尽可能减小通信开销；b）非支配排序和拥挤度计算是个体间关联度较大的操作，其计算在每个从核实现并行化时，从核需要获取全部个体信息，而申威处理器的主核内主存速度慢，从核内局存容量又过小；c）若考虑将数据分散储存在各个从核的局存中，数据的共享需依赖于各从核间的大量寄存器通信，而寄存器通信机制较为严格，需要发送与接收一一对应、配合有序，大量通信时需要具有完善的调度机制；d）从核寄存器通信虽快，但每次仅支持 1 个 256 位长度的向量，过量通信依然会导致通信开销占比增大，从而影响并行效果。针对并行中的这些问题，本文主要从寄存器通信、双缓冲、向量化以及存储优化四个方面进行并行优化。

2.3.1 寄存器通信

如图 4 所示，SW26010 从核阵列支持同行和同列的寄存器通信，但从核通信需遵守一定的规则：a）寄存器间通信时，从核接收信息次数与其他从核向其发送信息的次数应该相等，若接收次数大于发送次数，则从核会停滞，直到接收到新的信息，若一直未收到信息，则该从核将一直等待，造成整个程序无法进行；b）从核间的发送通信在一定数据量内是非阻塞的，当某一从核对另一从核发送信息时，先将信息放入从核所对应的物理缓冲区，使通信能够非阻塞进行，不用等待，但若通信量过大而未及时接收，物理缓冲区被占满时，从核将无法再发送信息，进入停滞状态；c）当两个从核同时向对方发送信息，两个从核均处于信息发送状态而没有接收信息，这会产生死锁，使程序无法进行。根据以上寄存器通信规则，本文对非支配排序和拥挤度计算进行并行优化。

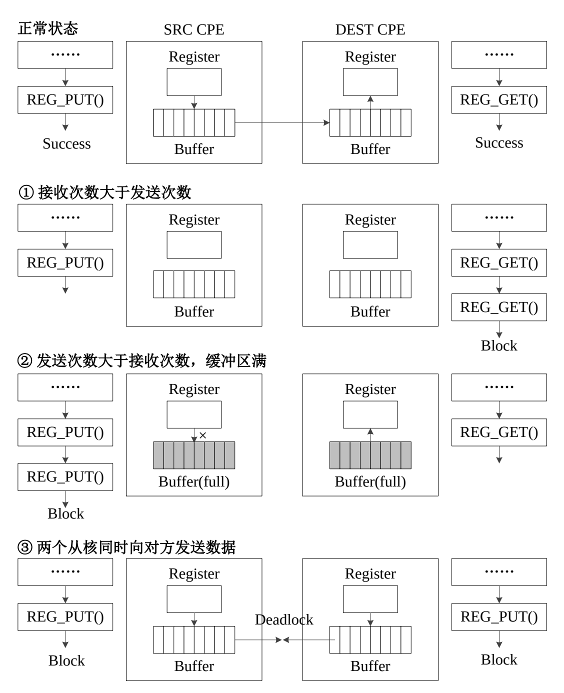


图 4　寄存器通信规则

Fig.4　Register communication rules

1）非支配排序的优化

在非支配排序过程中频繁访问主核主存使通信开销急剧增加，而每个从核 64 KB 的局存又无法完整存储所有个体的目标函数值，因此需要将所有个体数据划分成 64 等份，分散存储在每个从核局存中；另一方面，非支配排序需要所有个体的信息，从核间需要频繁进行数据通信，并且保证分层过程中的信息完整性。

本文设计了一条哈密顿回路，如图 5 所示。每个个体在每一层分层过程中经过其余所有从核的次数有且仅有一次，相邻

的两个从核可以沿固定方向通信，并且首尾相连形成环路。这种设计使每一个从核只向特定的从核发送数据，也只接收特定从核发来的数据，发送和接收的次数依照从核内处理的个体数而定，从而可以避免不确定的通信方向和通信次数导致的不必要的开销和寄存器通信异常。

从核进行支配关系的比较时，当前个体与本从核内其余个体比较后，传递到下一个从核，与下一个从核内各个体进行比较，直到同所有的从核内的所有个体完成比较。为了判断一次循环后所有个体是否被完全分层，需要统计每一个从核内未被分层的个体数量。从第一个从核开始，未分层个体数量依次向后发送并进行累加，直到第一个从核收到最后一个从核统计的个数。若已完全分层，则向所有从核广播停止指令；若未分层完毕，则继续下一次非支配排序。

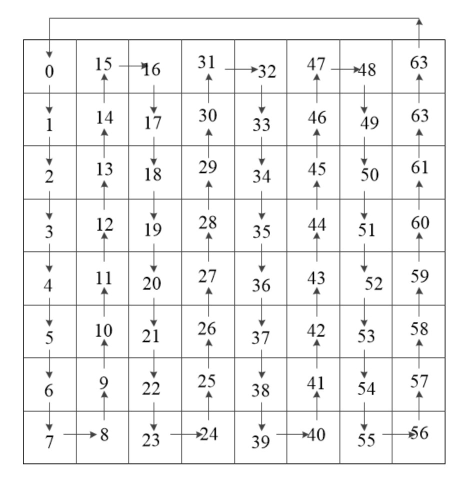


图 5 从核间通信的哈密顿回路

Fig.5 Hamiltonian loop of communication among CPEs

2）拥挤度计算的优化

拥挤度计算以非支配排序的分层结果为基础。分层结束后，每个从核只保存部分个体。拥挤度计算前要将每一层个体的每一个适应度值进行从大到小的排序，若将数据传回主核进行排序，之后再由从核进行距离计算，会影响并行效率。因此，本文使用从核通信哈密顿回路，在从核内直接进行目标函数值的排序。每个个体在各从核间第一轮传递时，分别与每个从核的所有个体比较，记录个体所在层中比该个体大的数量，当个体回到原从核，即可得到该个体在每一层中所排的序号。在第二轮传递时，计算每个个体与其序号相邻的两个个体的目标函数差值（除最大和最小的个体外）。最后所有个体回到原从核，对个体的每一个目标函数的差值进行累加计算出拥挤度距离。

以上优化中，利用从核间寄存器通信技术，设计严格的通信规则，实现了核组内从核局存的共享，解决了单个从核局存空间过小而无法存储全部所需数据的问题，提高了并行效率。

2.3.2 双缓冲

对于目标函数计算，在 DMA 传输的基础上加入双缓冲机制，可以实现计算和通信最大限度的互相隐藏，取得良好的加速效果。双缓冲机制就是在从核的局部数据存储上申请两倍于一次处理所需数据大小的空间，以便存放两份同样大小且互为对方缓冲的数据。通过编程来控制和实现。具体过程如下：除了处理第一轮次以及最后一轮次读入（写回）数据的通信过程之外，当计算核心进行本轮次数据计算的同时，进行下一轮次（上一轮次）读入（写回）数据的通信。

因为双缓冲机制的效果受通信和计算开销比例的影响，多次调用 DMA 传输需要额外的开销，并且所传输的数据必须是 32 KB 对齐且为连续内存块，所以一次传输的数据应该尽可能多且 32 KB 对齐。针对不同的计算任务需通过测试找到最优的每次通信的数据量。

2.3.3 向量化

SW26010 的主从核核心都支持 256 位宽的向量化操作。例如，一次可以操作处理四个 double 类型。在非支配排序过程中涉及到大量的适应度值大小比较，若将四个 double 型的目标函数值载入 256 位向量同时进行比较，则将在理论上提高四倍的处理速度。在目标函数计算时涉及到大量数据的累加，采用向量加操作可大幅提升效率。

向量化操作时要注意对界问题。因为不对界的 Load/Store 会引发异常，操作系统收到异常信号后会将这些 Load/Store 拆分成标准类型，会大幅降低性能。因此在利用向量化操作时，对于不对界的一组目标函数值最好进行补全填充或合理分割。

2.3.4 存储优化

为了方便和直观地进行信息的存储，种群中每个个体一般以结构体的形式存储。若在主从核间直接以结构体形式传输，则可能附带传输与本次计算无关的数据，不仅浪费从核局存且增大了通信开销。若只取所需要的部分数据，则会导致所取数据的存储方式是离散的，不仅导致 cache 命中率降低、计算开销急剧增加，且影响 DMA 传输速度。因此，考虑在目标函数计算前，先将个体中相关的解码后的信息连续存储，减小 DMA 通信的开销和从核内读取的开销。从核内计算完毕后，将适应度值依然存储在连续空间内传回主核，再进行分别赋值。

2.4 多核组并行

随着问题规模扩大，种群个体数增加。由于单核组能处理的种群规模数有限，所以本文使用 MPI 编程模型将单核组的并行 NSGA-II 扩展到大规模集群上，有利于 NSGA-II 在大规模问题上表现出较好的适用性。

2.4.1 分岛模型

鉴于 SW26010 的架构特点，多核组的并行采用分岛模型。分岛模型将初始种群按照核组（1 个主核+64 个从核）数量分成若干个子种群，各子种群独立地在每个岛上执行遗传操作（一个岛对应一个核组），如图 6 所示。每隔一定代数，各子种群间进行数据迁移，增加子种群的多样性，防止过早收敛。分岛模型通信开销较小，理论上可以获得接近线性的加速比。

2.4.2 迁移策略

为了防止产生局部最优解，每个孤岛每隔一定代数需要进行个体迁移。具体策略为：每个进程均向其他 n-1 个进程发送个体，进程 m 根据进程 id 和当前进化代数确定接收个体；迁

移时，按照分层序号进行迁移，迁移的个体为非支配排序的最外层个体，确保迁出的都是岛内优良的个体；接收时，将接收到的优良个体跟本岛内的层级靠后的个体进行替换。

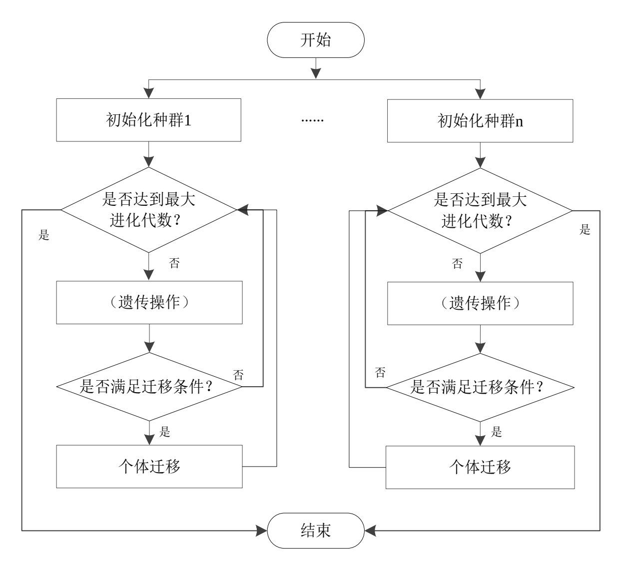


图 6　分岛模型流程

Fig.6　Flow chart of island model

3　实验结果与分析

本实验基于 SW26010 众核处理器，测试函数为 ZDT1 及 BinhKorn [25]，种群大小为 64 000，进化代数为 10，交叉概率为 0.8，变异概率为 0.1，从核数默认为 64。

ZDT1 函数：

$$f_1(x)=x_1 \tag{1}$$

$$f_2(x)=g(x)[1-\sqrt{\frac{x_1}{g(x)}}] \tag{2}$$

$$g(x)=1+\frac{9}{n-1}\sum_{i=2}^{n}x_i$$
$$x_i\in[0,1], i=1,2,\Lambda,n.2\le n\le 30; \tag{3}$$

BinhKorn 函数：

$$f_1(x,y)=4x^2+4y^2;0\le x\le 5; \tag{4}$$

$$f_2(x,y)=(x-5)^2+(y-5)^2;0\le y\le 3; \tag{5}$$

约束条件：

$$g_1(x,y)=(x-5)^2+y^2\le 25; \tag{6}$$

$$g_2(x,y)=(x-8)^2+(y+3)^2\ge 7.7; \tag{7}$$

为了衡量程序的并行化效果和从核利用率，本文引入了性能评价指标——从核加速比，其定义为同等条件下只使用主核的运行时间除以主核和从核协同工作的运行时间。

3.1　解的正确性

图 7 和 8 分别为 ZDT1 和 BinhKorn 采用串行 NSGA-II 和单核组并行 NSGA-II 所得 Pareto 解集。由图可得，两种方法所得的解集基本重合，保证了单核组并行 NSGA-II 的正确性，且并行 NSGA-II 的解集分布更均匀。

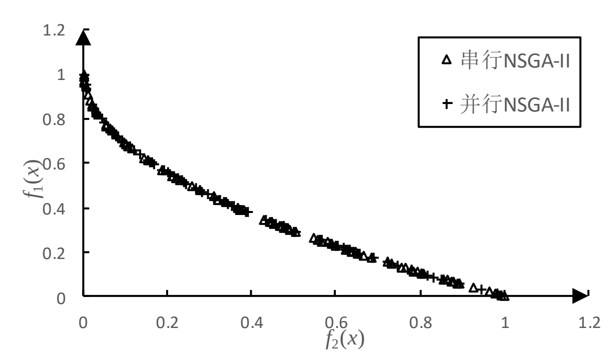


图 7　串行 NSGA-II 与并行 NSGA-II 所求 ZDT1 解集

Fig.7　Solutions of ZDT1 by serial NSGA-II and parallel NSGA-II

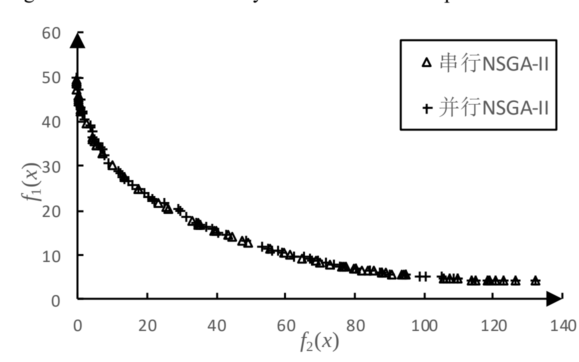


图 8　串行 NSGA-II 与并行 NSGA-II 所求 BinhKorn 解集

Fig.8　Solutions of BinhKorn by serial NSGA-II and parallel NSGA-II

3.2　加速比与扩展性

在保持种群总个体数不变的情况下增加核组数，所得的从核加速比曲线如图 9 所示。两种测试函数的从核加速比呈相同的趋势，均先增加后减小，峰值出现在核组数为 2 时，从核加速比可分别达到 13.95 和 16.28。为了分析目标函数计算、非支配排序以及拥挤度计算三个模块的并行效果对算法整体从核加速比的影响，表 1 和 2 分别列出了采用 ZDT1 和 BinhKorn 两种测试函数的并行 NSGA-II 的三个模块的运行时间和从核加速比，以及算法整体的从核加速比。其中，非支配排序的从核加速比最高，而目标函数计算的从核加速比最低。由于非支配排序占总运行时间比重最多，所以算法整体的从核加速比的走势基本与非支配排序的从核加速比一致。与 ZDT1 相比，由于 BinhKorn 函数加入了约束条件的限制，非支配排序运行时间占比更大，所以其算法整体的从核加速比更高且更贴近非支配排序的从核加速比。

在实验中，由于从核局存较小，从核处理个体的数量有限，且测试函数的计算量过小而相对通信开销较大，导致了目标函数计算的从核加速比小于 1。这也体现了轻载时对非支配排序和拥挤度计算进行并行化的必要性和重要性。但如果不对目标函数计算进行并行，则会显著增加非支配排序的通信开销，权衡两部分的加速效果，有必要对目标函数进行并行化。

表1　采用ZDT1的并行NSGA-II运行时间和加速比

Table 1　Run time and speedups of parallel NSGA-II with ZDT1

核组数	目标函数计算			非支配排序			拥挤度计算			从核
	主核串行时间	从核并行时间	从核加速比	主核串行时间	从核并行时间	从核加速比	主核串行时间	从核并行时间	从核加速比	加速比
1	6.10E+06	7.32E+06	0.83	3.61E+11	1.66E+10	21.78	3.93E+10	6.26E+09	6.28	12.10
2	3.02E+06	3.72E+06	0.82	8.83E+10	3.42E+09	**25.84**	1.04E+10	1.58E+09	6.55	**13.95**
4	1.33E+06	1.84E+06	0.75	1.92E+10	7.48E+08	25.72	2.69E+09	3.99E+08	6.75	13.76
8	6.49E+05	1.15E+06	0.60	3.97E+09	1.55E+08	25.67	7.31E+08	1.03E+08	7.06	13.30
16	3.19E+05	5.67E+05	0.56	9.13E+08	3.84E+07	23.75	2.05E+08	2.73E+07	7.50	12.28
32	1.53E+05	2.91E+05	0.49	1.87E+08	9.15E+06	20.44	5.65E+07	7.13E+06	7.93	11.08
64	6.59E+04	1.50E+04	0.44	4.28E+07	2.32E+06	18.42	1.59E+07	1.88E+06	**8.49**	9.14

表2　采用BinhKorn的并行NSGA-II运行时间和加速比

Table 2　Run time and speedups of parallel NSGA-II with BinhKorn

核组数	目标函数计算			非支配排序			拥挤度计算			从核
	主核串行时间	从核并行时间	从核加速比	主核串行时间	从核并行时间	从核加速比	主核串行时间	从核并行时间	从核加速比	加速比
1	6.11E+06	7.33E+06	0.83	5.05E+11	2.32E+10	21.77	3.93E+10	6.25E+09	6.30	14.69
2	3.02E+06	3.72E+06	0.81	1.24E+11	4.78E+09	**25.83**	1.04E+10	1.55E+09	6.68	**16.28**
4	1.33E+06	1.85E+06	0.72	2.69E+10	1.07E+09	25.13	2.68E+09	3.52E+08	7.62	16.06
8	6.49E+05	1.07E+06	0.61	5.47E+09	2.24E+08	24.41	7.25E+08	9.25E+07	7.84	15.67
16	3.19E+05	5.64E+05	0.57	1.22E+09	5.13E+07	23.75	2.03E+08	2.54E+07	7.98	14.39
32	1.53E+05	2.93E+05	0.52	2.61E+08	1.27E+07	20.52	5.86E+07	7.19E+06	8.15	12.01
64	7.40E+04	1.61E+05	0.46	6.37E+07	3.62E+06	17.59	1.65E+07	1.93E+06	**8.56**	9.48

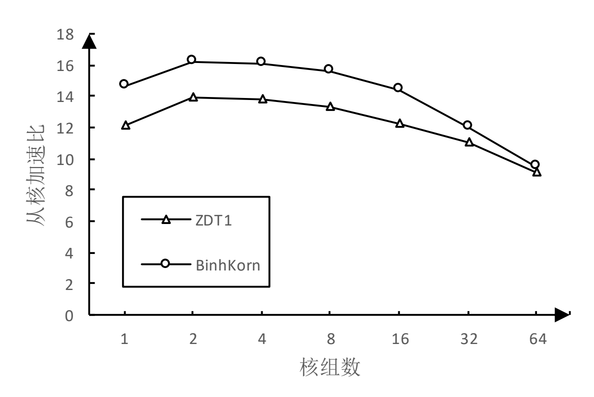


图9　两种测试函数的从核加速比

Fig.9　Slave-core speedups with two test functions

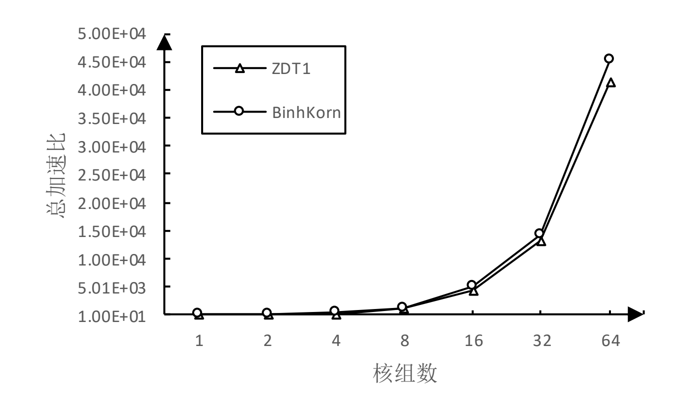


图10　两种测试函数的总加速比

Fig.10　Speedups with two test functions

非支配排序与其他两个模块相比加速效果较好，从核加速比最高可超过25。随着核组数量的增加，每个从核所分配到的需要处理的个体数减少，从核计算量也随之减少，从而导致加速效果下降。但从核处理个体数过多时，也会导致在分层过程中各从核间由于随机性带来的计算量差异增大，从而使寄存器通信过程中等待的时间增加，影响从核加速比。所以当核组数较少时，从核加速比反而会下降。

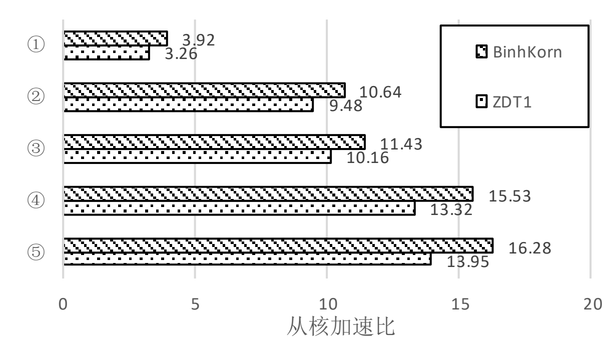
对于拥挤度计算，在串行程序中，个体先要与其所在的非支配层内其他个体进行排序。而在并行程序中，由于同一非支配层的个体可能分布在不同的从核内，所以个体需要与种群内每一个个体进行所属层级的确认，再对同层内的个体进行排序。当核组数越多，孤岛内子种群个体数越少，虽然从核的计算量也随之减少，但拥挤度排序过程中与非本层内个体比较的额外开销减少更多，因此从核加速比呈上升趋势。

多核组实验的总加速比如图10所示。图中可看出，随着核数的增加，加速比呈次方级增大，并行效率远远超过了1，最终可在64核组条件下达到超过40 000的加速比。这是因为算法中非支配排序所占的开销最大，相比于其余部分的开销多出2-3个数量级，而非支配排序的时间复杂度为O(N^2)，N为每个孤岛的个体数量。在种群个体总数保持不变的情况下，孤岛数量越多，每个孤岛所分到的个体数量越少，时间开销则呈个体数量减小的次方级减小。由此可得，申威众核处理器在利用NSGA-II解决大规模问题上有较好的扩展性。



3.3 优化方法对比

如图 11 所示，实验采用两个核组，寄存器通信的使用使从核加速比提升显著，相对于未优化的并行 NSGA-II，分别提升了 190.8%和 171.4%。此外，向量化也具有较好的优化效果，相较于未加入向量化方法，加速比提升了 31.1%和 35.9%。



注：①并行 NSGA-II；

②并行 NSGA-II+优化（寄存器通信）；

③并行 NSGA-II+优化（寄存器通信+双缓冲）；

④并行 NSGA-II+优化（寄存器通信+双缓冲+向量化）；

⑤并行 NSGA-II+优化（寄存器通信+双缓冲+向量化+存储优化）。

图 11　不同优化方法对从核加速比的影响

Fig.11　Influence of different optimization methods on slave-core speedup

4　结束语

本文基于申威众核处理器的体系架构特点，设计和实现 了“分岛—主从增强”混合式并行 NSGA-II 算法，并综合运用了多种并行优化方法。这些设计思路和优化方法可为其他算法在太湖之光上的并行和优化提供借鉴。本文主要贡献如下：

a）在主从模式的基础上，利用从核间寄存器通信，实现核组内从核局部数据存储的共享，提出了“主从增强”模式的并行。

b）优化算法流程，实现更多算法模块在从核上的并行，提高了算法的并行化程度。

c）综合运用 DMA 传输、向量化、双缓冲、存储优化等优化方法显著提高加速比，解决了轻载情况下加速效果较差的问题，保证了算法在多种负载情况下并行效率。

d）利用分岛模型和迁移策略，将算法扩展到申威大规模集群上，证明了其良好的扩展性。

实验证明了算法的正确性。在对 ZDT1 和 BinhKorn 两个测试函数的实验中，从核加速比可分别达到 13.95 和 16.28。当扩展到 64 个核组时，总加速比可超过 40 000。

然而受限的从核局存容量为算法设计带来了巨大挑战，进一步优化访存，提高访存效率和 Cache 命中率是未来工作的重点。

参考文献：

[1] 杨广文，赵文来，丁楠，等. “神威 • 太湖之光”及其应用系统 [J]. 科学, 2017 (3): 12-16. (Yang Guangwen, Zhao Wenlai, Ding Nan, *et al.* ” Sunway TaihuLight” and its application system [J]. Science, 2017 (3): 12-16.)

[2] 刘鑫，郭恒，孙茹君，等. “神威 • 太湖之光”计算机系统大规模应用特征分析与 E 级可扩展性研究 [J/OL]. 计算机学报, 2018, 41 (24): 1-10. http://kns. cnki. net/kcms/detail/11. 1826. TP. 20180303. 1356. 012. html. (Liu Xin, Guo Heng, Sun RuJun, *et al.* The characteristic analysis and exascale scalability research of large scale parallel applications on sunway taihuLight supercomputer [J/OL]. Chinese Journal of Computers, 2018, (41) (24): 1-10.)

[3] 洪文杰，李肯立，全哲，等. 面向神威 • 太湖之光的 PETSc 可扩展异构并行算法及其性能优化 [J]. 计算机学报, 2017 (9): 2057-2069. (Hong Wenjie, Li Kenli, Quan Zhe, *et al.* PETSc's heterogeneous parallel algorithm design and performance optimization on the sunway TaihuLight system [J]. Chinese Journal of Computers, 2017 (9): 2057-2069.)

[4] 孟德龙，文敏华，韦建文，等. 神威太湖之光上 OpenFOAM 的移植与优化 [J]. 计算机科学, 2017 (10): 64-70. (Meng Delong, Wen Minhua, Wei Jianwen, *et al.* Porting and optimizing OpenFOAM on sunway TaihuLight system [J]. Computer Science, 2017 (10): 64-70.)

[5] 赵美婷，刘轶，刘锐，等. 基于申威众核处理器的 HOG 特征提取算法并行加速 [J]. 计算机工程与科学, 2017 (4): 611-618. (Zhao Meiting, Liu Yi, Liu Rui, *et al.* Acceleration of histogram of oriented gradient (HOG) based on sunway many-core processor [J]. Computer Engineering and Science, 2017 (4): 611-618.)

[6] 姚文军，陈俊仕，苏志超，等. 基于神威太湖之光的 NAMD 软件的移植与优化 [J]. 计算机工程与科学, 2017 (6): 1022-1030. (Yao Wenjun, Chen Junshi, Su Zhichao, *et al.* Porting and optimizing of NAMD on sunway TaihuLight system [J]. Computer Engineering and Science, 2017 (6): 1022-1030.)

[7] 姜雪松，江树刚，张玉，等. 国产众核处理器百万核时域有限差分并行计算 [J]. 西安电子科技大学学报：自然科学版, 2017 (6): 65-69, 128. (Jiang Xuesong, Jiang Shugang, Zhang Yu, *et al.* Parallel FDTD computation of million cores with the domestically-made many-core supercomputer [J]. Journal of Xidian University: Natural Science, 2017 (6): 65-69, 128.)

[8] 何王全，刘勇，方燕飞，等. 面向国产异构众核系统的 Parallel C 语言设计与实现 [J]. 软件学报, 2017 (4): 764-785. (He Wangquan, Liu Yong, Fang Yanfei, *et al.* Design and implementation of parallel C programming language for domestic heterogeneous many-core systems [J]. Journal of Software, 2017 (4): 764-785.)

[9] 敖玉龙. 国产大型众核系统上稀疏矩阵和 Stencil 运算的性能优化关键技术研究 [D]. 北京：中国科学院大学, 2017. (Ao Yulong. Research on key optimizations of sparse matrix and stencil computation for the domestic large many-core system [D]. Beijing: University of Chinese Academy of Sciences, 2017.)

[10] Sara K, Mohammad T. An improved parallel genetic algorithm for optimal sensor placement of wireless sensor networks [C]// Proc of ACM



Conference on Wireless Network Security. Switzerland: [s. n.] , 2014: 261-268.

[11] Sanhueza C, Jiménez F, Berretta R, *et al.* PasMoQAP: a parallel asynchronous memetic algorithm for solving the multi-objective quadratic assignment problem [C]// Proc of Evolutionary Computation. Spain: IEEE Press, 2017: 1103-1110.

[12] Potuzak T. Distributed//parallel genetic algorithm for road traffic network division using a hybrid island model//step parallelization approach [C]// Proc of IEEE//ACM International Symposium on Distributed Simulation and Real Time Applications. London: IEEE Press, 2016: 170-177.

[13] Kseniya N, Alexey R. A parallel genetic algorithm approach for monitoring devices placement [C]// Proc of International Multi-Conference on Engineering, Computer and Information Sciences. 2017: 186-189.

[14] Ortega G, Filatovas E, Garzón E M, *et al.* Non-dominated sorting procedure for Pareto dominance ranking on multicore CPU and//or GPU [J]. Journal of Global Optimization, 2017, 69: 1-21.

[15] 苏兆品，张国富，蒋建国，等. 基于非支配排序差异演化的应急资源多目标分配算法 [J]. 自动化学报, 2017 (2): 195-214. (Su Zhaopin, Zhang Guofu, Jiang Jianguo, *et al.* Multi-objective approach to emergency resource allocation using none-dominated sorting based differential evolution [J]. Acta Automatica Sinica, 2017 (2): 195-214.)

[16] 陈志旺，黄兴旺，陈志兴，等. 区间多目标优化非支配排序云模型算法 [J]. 计算机工程与应用, 2016, 53 (22): 143-149. (Chen Zhiwang, Huang Xingwang, Chen Zhixing, *et al.* Non-dominated sorting cloud model algorithm for inter-val multi-objective optimization [J]. Computer Engineering and Applications, 2016, 53 (22): 143-149.)

[17] 邓凯文，韩肖清，梁琛. 精英非支配排序算法与改进粒子群算法相结合的储能优化配置 [J]. 科学技术与工程，2017 (20): 171-177. (Deng Kaiwen, Han Xiaoqing, Liang Chen. Optimal configuration of energy storage based on elitist non-dominated sorting genetic algorithm Ⅱ with improved multi-objective particle swarm optimization [J]. Science Technology and Engineering, 2017 (20): 171-177.)

[18] 赵瑞祥，郑凯，刘垚，等. 基于申威众核处理器的混合并行遗传算法 [J]. 计算机应用, 2017 (9): 2518-2523. (Zhao Ruixiang, Zheng Kai, Liu Yao, *et al.* Hybrid parallel genetic algorithm based on sunway many-core processors [J]. Journal of Computer Applications, 2017 (9): 2518-2523.)

[19] Zhang Wenqiang, Fujimura Shigeru. Multiobjective process planning and scheduling using improved vector evaluated genetic algorithm with archive [J]. IEEJ Transactions on Electrical and Electronic Engineering, 2012 (3): 258-267.

[20] Zhang Rui, Chiong R. Solving the energy-efficient Job-Shop scheduling problem: a multi-objective genetic algorithm with enhanced local search for minimizing the total weighted tardiness and total energy consumption [J]. Journal of Cleaner Production, 2016, 112 (1): 3361-3375.

[21] Mousa A A, Elattar E E. Best compromise alternative to EELD problem using hybrid multiobjective quantum genetic algorithm [J]. Applied Mathematics & Information Sciences, 2014, 8 (6): 2889-2902.

[22] Deb K, Agrawal S, Pratap A, *et al.* A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II [C]// Proc of Parallel Problem Solving From Nature Vi Conference. Berlin: Springer, 2000: 849-858.

[23] Sheng Wanxing, Liu Keyan, Liu Yuan, *et al.* Optimal placement and sizing of distributed generation via an improved nondominated sorting genetic algorithm II [J]. IEEE Trans on Power Delivery, 2015, 30 (2): 569-578.

[24] 沈焕学. NSGA-II 并行化研究在国产众核上的实现 [D]. 上海：华东师范大学, 2018. (Shen Huanxue. Parallel research of NSGA-II and implement on domestic many-core processor [D]. Shanghai: East China Normal University, 2018.)

[25] Deb K, Thiele L, Laumanns M, *et al.* Scalable test problems for evolutionary multiobjective optimization [M]. London: Springer, 2005: 105-145.